

(12) DEMANDE INTERNATIONALE PUBLIÉE EN VERTU DU TRAITÉ DE COOPÉRATION  
EN MATIÈRE DE BREVETS (PCT)

(19) Organisation Mondiale de la Propriété  
Intellectuelle  
Bureau international



(43) Date de la publication internationale  
22 septembre 2005 (22.09.2005)

PCT

(10) Numéro de publication internationale  
**WO 2005/088928 A1**

(51) Classification internationale des brevets<sup>7</sup> :  
**H04L 29/06**, H04N 7/26, G06F 17/30

(21) Numéro de la demande internationale :  
PCT/FR2004/000340

(22) Date de dépôt international :  
13 février 2004 (13.02.2004)

(25) Langue de dépôt : français

(26) Langue de publication : français

(71) Déposant (pour tous les États désignés sauf US) :  
**FRANCE TELECOM** [FR/FR]; 6, place d'Alleray,  
F-75015 Paris (FR).

(72) Inventeur; et

(75) Inventeur/Déposant (pour US seulement) : **GEGOUT,**

**Cédric** [FR/FR]; 5, rue Monseigneur Duchesne, F-35000  
Rennes (FR).

(74) Mandataires : **LOUISET, Raphaël** etc.; Cabinet Plasser-  
aud, 65/67, rue de la Victoire, F-75440 Paris Cedex 9 (FR).

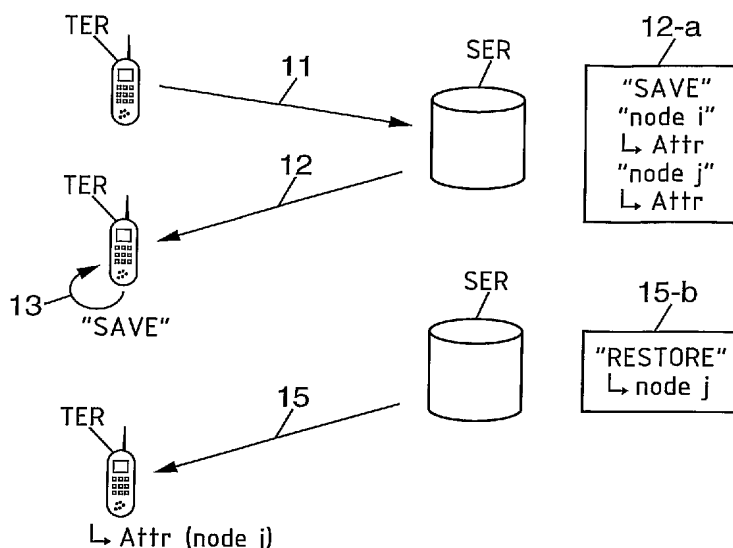
(81) États désignés (sauf indication contraire, pour tout titre de  
protection nationale disponible) : AE, AG, AL, AM, AT,  
AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO,  
CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB,  
GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG,  
KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG,  
MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM, PG, PH,  
PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SY, TJ, TM, TN,  
TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

(84) États désignés (sauf indication contraire, pour tout titre de  
protection régionale disponible) : ARIPO (BW, GH, GM,  
KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), eurasien

[Suite sur la page suivante]

(54) Title: METHOD FOR EDITING MULTIMEDIA PAGES ON A TERMINAL USING PRE-STORED PARAMETERS OF  
OBJECTS APPEARING IN SCENES

(54) Titre : PROCEDE D'EDITION DE PAGES MULTIMEDIA AUPRES D'UN TERMINAL, AVEC PRE-MEMORISATION  
DE PARAMETRES D'OBJETS INTERVENANT DANS LES SCENES.



(57) Abstract: The invention relates to multimedia page editing via a terminal. A server supplies multimedia pages to the terminal in the form of object arrangement commands for objects identified by parameters. A method according to the invention comprises (a) a preliminary step (12) wherein the server transmits all or part of the object-related parameters as well as a store command whereby the parameters are stored in a terminal memory; and (b) a main step (15) wherein the server transmits a simple parameter retrieval command for editing at least one multimedia page in which the above-mentioned object appears.

[Suite sur la page suivante]

WO 2005/088928 A1



(AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), européen (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

*En ce qui concerne les codes à deux lettres et autres abréviations, se référer aux "Notes explicatives relatives aux codes et abréviations" figurant au début de chaque numéro ordinaire de la Gazette du PCT.*

**Déclaration en vertu de la règle 4.17 :**

— *relative à la qualité d'inventeur (règle 4.17.iv)) pour US seulement*

**Publiée :**

— *avec rapport de recherche internationale*

---

**(57) Abrégé :** L'invention concerne l'édition de pages multimédia auprès d'un terminal. Un serveur distribue au terminal des pages multimédia sous forme d'instructions d'agencements d'objets identifiés par des paramètres. Un procédé au sens de l'invention comporte : a) une étape préalable (12) pendant laquelle le serveur transmet au moins une partie des paramètres associés à un objet, ainsi qu'une instruction de stockage de ces paramètres dans une mémoire du terminal, b) et une étape courante (15) pendant laquelle le serveur transmet une instruction de simple restauration des ces paramètres, pour éditer au moins une page multimédia dans laquelle intervient l'objet précité.

Procédé d'édition de pages multimédia auprès d'un  
terminal, avec pré-mémorisation de paramètres d'objets  
intervenant dans les scènes

5 L'invention concerne l'édition de pages multimédia auprès de terminaux, notamment dans le cadre de services multimédia proposés sur des téléphones mobiles agencés pour coopérer avec des réseaux cellulaires.

10 Dans le contexte de l'invention, un serveur distribue, à un ou plusieurs terminaux, une partie au moins de pages multimédia sous forme d'instructions d'agencements d'objets intervenant dans une page multimédia et identifiés par des paramètres associés.

15 On entend par "*page multimédia*" par exemple une scène graphique à éditer auprès du terminal, le cas échéant agrémentée d'une ou plusieurs séquences sonores à jouer sur des baffles ou écouteurs du terminal.

20 Dans ce contexte, un même objet (par exemple un objet graphique dans une scène graphique) peut être utilisé d'une page à l'autre ou un même objet peut conserver les mêmes paramètres d'agencements d'une page multimédia à  
25 l'autre. Dans ce cas, il se pose le problème de transmission et stockage systématiques et inutiles de données relatives à cet objet ou encore à des paramètres d'agencements du même objet, pour plusieurs pages dans lesquelles le même objet intervient avec les mêmes  
30 paramètres d'agencements. Ce problème devient particulièrement gênant lorsque l'on doit avoir recours à

plusieurs échanges entre le terminal et le serveur d'autant plus que la bande passante allouée pour la communication entre le serveur et le terminal (notamment mobile) peut être restreinte.

5

La présente invention propose un mécanisme de stockage d'informations sur les objets qui apparaissent dans la description d'une page multimédia.

10

Pour ce qui concerne l'édition de scènes graphiques, plusieurs formats de représentation graphique d'animations graphiques existent actuellement. Cependant, aucun de ces formats ne propose un tel mécanisme de stockage.

15

On connaît des techniques qui permettent de stocker les informations partielles d'une scène graphique, mais qui utilisent à cet effet des méthodes programmatiques (par exemple les formats MPEG/MPEGJ ou VRML/EAI), dans un but tout différent de celui de la présente invention.

20

D'ailleurs, ces techniques souffrent d'un manque de souplesse, en ce sens qu'il est impossible de télécharger des morceaux de contenu programmatique. Elles souffrent aussi d'un manque d'efficacité dans le rendu graphique, en ce sens qu'il est nécessaire de lancer une machine

25

virtuelle pour traiter le contenu programmatique.

30

L'un des buts visés par la présente invention concerne la réduction de la mémoire, notamment graphique, des terminaux, nécessaire à l'édition de pages multimédia complexes, ou d'une succession de telles pages.

Un autre but visé par la présente invention concerne la réduction des ressources de calcul nécessaires à l'édition de telles pages, ou d'une succession de ces pages.

5 Un autre but visé par la présente invention est de fournir un procédé permettant d'accomplir les buts visés ci-avant tout en offrant une compatibilité avec les techniques classiques de décodage.

10 Plus généralement, un but visé par la présente invention est d'offrir une plus grande flexibilité au niveau des requêtes et données échangées entre le serveur et le terminal.

15 La présente invention propose tout d'abord un procédé dans lequel un serveur distribue, à un ou plusieurs terminaux, une partie au moins de pages multimédia sous forme d'instructions d'agencements d'objets identifiés par des paramètres associés,

20 ce procédé comportant :

a) au moins une étape préalable pendant laquelle le serveur transmet au moins une partie des paramètres associés à un objet, ainsi qu'une instruction de stockage desdits paramètres dans une mémoire du  
25 terminal,

b) et au moins une étape courante pendant laquelle le serveur transmet une instruction de simple restauration desdits paramètres préalablement stockés en mémoire du terminal, pour éditer au moins une page multimédia dans  
30 laquelle intervient ledit objet.

Ainsi, le procédé au sens de l'invention permet une réduction de la mémoire du terminal, notamment de sa mémoire graphique par exemple dans le contexte d'édition de scènes graphiques, puisque l'on ne mémorise que la  
5 commande de stockage, par exemple d'informations sur des nœuds graphiques décrivant des objets intervenant dans une ou plusieurs scènes.

Le procédé au sens de l'invention permet aussi un gain  
10 dans l'utilisation des ressources de calcul, puisque, typiquement, l'utilisation d'un contenu programmatique tel qu'il est proposé dans MPEG-4/System/MPEGJ (ou en SVG/DOM) induirait un net surcoût de calcul, au moins pour certaines animations. Avantageusement, le procédé au sens  
15 de l'invention, utilisant un mécanisme conforme à un processus classique de rendu de commande graphique ou sonore, est alors facile à implémenter, en particulier pour un système à terminaux mobiles.

20 Le procédé au sens de l'invention offre aussi une compatibilité avec les techniques classiques de décodage, puisque le procédé peut être mis en œuvre dans la plupart des dispositifs de rendu graphique et/ou sonore.

25 Selon une caractéristique avantageuse, l'étape b) seule est réitérée pour l'édition de plusieurs pages multimédia dans lesquelles intervient ledit objet.

Dans une réalisation, les paramètres stockés comportent au  
30 moins des attributs déclaratifs d'un agencement de l'objet

dans une ou plusieurs pages multimédia dans lesquelles cet objet intervient avec les mêmes attributs.

Préférentiellement, ces paramètres comportent en outre un  
5 identifiant d'une zone mémoire du terminal allouée pour le  
stockage des attributs et, avantageusement, l'instruction  
de restauration comporte l'identifiant de cette zone  
mémoire pour retrouver les attributs précités.

10 Dans un mode de réalisation préféré, le procédé comporte  
en outre une étape de fin d'édition de pages multimédia  
incluant l'objet précité, étape dans laquelle le serveur  
transmet au terminal une instruction d'effacement des  
paramètres associés à cet objet.

15 Avantageusement, cette instruction d'effacement comporte  
l'identifiant de la zone mémoire du terminal stockant les  
paramètres associés audit objet, pour effacer de cette  
zone mémoire ces paramètres.

20 Dans une réalisation avantageuse, l'objet précité est un  
objet graphique comportant au moins l'un des éléments  
parmi :

- une image,
- 25 - une séquence d'images,
- une séquence d'images synthétiques 2D  
(bidimensionnelles),
- et une séquence d'images synthétiques 3D  
(tridimensionnelles).

30

On indique que de telles séquences d'images sont susceptibles d'être utilisées par exemple par le standard MPEG-4.

5 Dès lors que les instructions précitées, en particulier de stockage et de restauration, apparaissent individuellement comme des moyens essentiels pour mettre en œuvre le procédé ci-avant, la présente invention vise aussi, d'abord, un produit programme sous forme de code  
10 informatique, et comportant une instruction de stockage, dans une mémoire d'un terminal, de paramètres d'au moins un objet destiné à être agencé, en fonction desdits paramètres, dans une page multimédia propre à être éditée auprès dudit terminal. La présente invention vise aussi un  
15 signal comportant ce code. Ce signal et/ou le produit programme lui-même, peuvent être transmis du serveur au terminal, ou encore émaner d'un support mémoire qui coopère avec un lecteur du terminal (tel qu'un lecteur de CD-ROM ou autre).

20

La présente invention vise aussi un produit programme sous forme de code informatique, et comportant une instruction de restauration de paramètres préalablement stockés dans une mémoire d'un terminal, ces paramètres étant associés à  
25 au moins un objet destiné à être agencé, en fonction desdits paramètres, dans une page multimédia propre à être éditée auprès dudit terminal. La présente invention vise aussi un signal comportant ce code. Ce signal et/ou le produit programme lui-même, peuvent être transmis du  
30 serveur au terminal, ou encore émaner d'un support mémoire



qui coopère avec un lecteur du terminal (tel qu'un lecteur de CD-ROM ou autre).

Enfin, la présente invention vise aussi un produit  
5 programme sous forme de code informatique, et comportant  
une instruction d'effacement de paramètres préalablement  
stockés dans une mémoire d'un terminal et associés à au  
moins un objet propre à être agencé, en fonction desdits  
paramètres, dans une page multimédia éditée auprès dudit  
10 terminal. La présente invention vise aussi un signal  
comportant ce code. Ce signal et/ou le produit programme  
informatique lui-même, peuvent être transmis du serveur au  
terminal, ou encore émaner d'un support mémoire qui  
coopère avec un lecteur du terminal (tel qu'un lecteur de  
15 CD-ROM ou autre).

D'autres caractéristiques et avantages de l'invention  
apparaîtront à l'examen de la description détaillée ci-  
après, et des dessins annexés sur lesquels :

- 20 - la figure 1 illustre les échanges entre un serveur SER  
et un terminal TER, pour le déroulement des étapes du  
procédé au sens de l'invention,
- la figure 2 représente schématiquement et partiellement  
les éléments d'un terminal TER,
- 25 - la figure 3 représente des agents logiciels  
interagissant pour l'édition de pages multimédia auprès du  
terminal TER.

En annexe, on a retranscrit les codes informatiques (en  
30 format binaire), respectifs des commandes "SAVE" (commande  
de stockage précitée), "RESTORE" (commande de restauration

précitée) et "CLEAN" (commande d'effacement précitée). Il faut comprendre que la description et son annexe en particulier présentent des caractéristiques susceptibles de contribuer à la définition de l'invention.

5

En se référant à la figure 1, le contexte d'application de l'invention peut être décrit par les étapes suivantes :

- Le terminal mobile TER demande une ou plusieurs pages multimédia définissant par exemple un contenu d'animation graphique, à un serveur SER (étape 11).
- Le serveur SER renvoie un contenu qui décrit l'agencement spatio-temporel des objets graphiques intervenant dans l'animation graphique (étape 12).
- Dans ce contenu, une fonction de stockage "SAVE" (correspondant à l'instruction de stockage précitée) est décrite dans la table 12-a. Cette fonction indique au terminal TER qu'il doit stocker des paramètres relatifs à différents objets susceptibles d'intervenir dans des futures scènes graphiques à construire. Ces objets sont identifiés par des nœuds graphiques i, j auxquels on associe notamment des attributs propres (Attr).
- Lorsque le terminal reçoit cette commande de stockage "SAVE", la "valeur" de l'objet graphique (notamment de ses attributs) est stockée en mémoire du terminal mobile (étape 13).
- Si le terminal TER reçoit ensuite (étape 15) une commande "RESTORE" (correspondant à l'instruction de restauration précitée), au moment où il rend la scène graphique, il doit exécuter cette commande. Le terminal

récupère les informations de l'objet graphique stocké pour les copier dans la scène graphique actuelle.

Bien entendu, si des paramètres associés à un objet graphique ne sont plus utiles pour l'édition de scènes graphiques suivantes, le serveur peut envoyer au terminal  
5 une commande "CLEAN" pour effacer ces paramètres de la mémoire du terminal.

Ainsi, ces commandes sont utilisées pour modifier un  
10 ensemble de propriétés d'une scène à un instant donné. Les commandes qui doivent être exécutées au même instant sont préférentiellement regroupées dans un même paquet (par exemple un paquet AccessUnit en MPEG-4/System, ou encore un paquet RTP). Dans le but de modifier la scène, le  
15 serveur doit donc transmettre des paquets qui contiennent une ou plusieurs de ces commandes.

On se réfère à la figure 2 pour décrire brièvement des modules prévus classiquement dans un terminal TER. Le  
20 terminal comporte un module de communication 21, notamment avec le serveur SER, duquel il reçoit les commandes précitées. Typiquement, la commande SAVE comporte un identifiant d'une zone mémoire ZMi ( $i=1,2,\dots$ ) dans une mémoire MEM prévue dans le terminal TER, pour stocker les  
25 paramètres associés à la commande SAVE dans cette zone mémoire ZMi. Ainsi, une commande suivante RESTORE, comportant le même identifiant de cette zone mémoire ZMi, permettra de récupérer ces paramètres de la mémoire MEM du terminal, ces paramètres étant traités dans une mémoire de  
30 travail 22 (par exemple par un logiciel du terminal dit "PLAYER"), éventuellement distincte de la mémoire MEM. Les

informations relatives aux objets de la scène en construction sont alors transférées vers une interface 23, par exemple une interface graphique pour un affichage de la scène (ou encore par une interface sonore pour un rendu  
5 sonore). De la même manière, la commande "CLEAN" comporte un identifiant de la zone mémoire ZMi pour effacer ensuite les paramètres stockés dans cette zone et qui ne sont plus utiles.

10 Ci-après, on définit la sémantique des commandes suivantes : "SAVE", "RESTORE" et "CLEAN".

La commande "SAVE" permet de stocker en mémoire certains attributs (ou une arborescence d'attributs) d'un nœud (ou  
15 objet graphique) contenu dans la scène courante. Un nœud sauvé reste en mémoire de manière permanente.

Les noeuds suivants peuvent être stockés :

- *Text* : les attributs *couleur* et *string* (chaîne  
20 de caractères) sont stockables par exemple pour un texte à afficher.

Après une commande SAVE, le nœud considéré peut  
avantageusement ne pas être stocké si la mémoire MEM du  
25 terminal est totalement remplie.

En se référant à l'annexe retranscrivant le code informatique de la commande "SAVE", les attributs *serviceID* et *groupID* spécifient une zone mémoire ZMi ou  
30 l'information doit être stockée.

Plus particulièrement, l'attribut `serviceID` est un entier indiquant le service ou la référence d'un contenu (référence de l'objet graphique à éditer), tandis que l'entier `groupID` indique l'espace de nommage des nœuds associés à cet objet. Enfin, le paramètre "nodes" correspond à une liste de nœuds dans laquelle chaque nœud est référencé par son nom (ou son identifiant ID).

La commande `RESTORE` (retranscrite en annexe) permet de récupérer des nœuds qui ont été stockés auparavant par la commande `SAVE`. Les nœuds récupérés vont remplacer les nœuds courants dans la scène courante suivant leur nom ou leur identifiant ID.

La commande `CLEAN` (retranscrite en annexe) permet d'effacer la zone mémoire identifiée par les attributs *service ID* et *groupID*.

Plus spécifiquement encore, les codes informatiques retranscrits en annexe sont en langage SDL (pour "Synthetic Description Language"). Ce langage est adopté pour définir les formats de *bitstream* (flux binaire de données). En particulier, des octets sont associés à chaque champ, comme on le verra en détail ci-après. Des informations complémentaires quant à ce langage peuvent être obtenues dans le descriptif de la norme ISO IEC 144-96.

On indique ici que la commande `SAVE` est déclarée par l'instruction `const bit(4) 9` qui signifie que la commande `SAVE` est déclarée par l'entier "9" qui devra se retrouver

dans les quatre premiers bits du flux binaire que reçoit le terminal. Ainsi, si ces quatre premiers bits comportent l'entier "9", ils déclareront la commande SAVE.

5 La commande suivante "uint(12) serviceID" définit le contexte dans lequel est identifiée un nœud graphique (ou sonore) d'une page multimédia. Plus particulièrement, la commande "unit(i)" sert à déclarer un entier non signé de i bits. On indique que "serviceID" peut viser un site  
10 multimédia, par exemple. La commande suivante déclare un paramètre "groupID" qui est relatif à une page multimédia par exemple du site "serviceID". Enfin, la commande déclarant "nbIds" sert à indiquer le nombre d'objets (graphiques ou sonores), identifiés par "id[i]" que l'on  
15 va sauvegarder dans la mémoire du terminal.

Ensuite, la commande "RESTORE" est déclarée par l'entier "10", dans les quatre premiers bits du flux binaire. Comme les identifiants "serviceID" et "groupID" sont aussi  
20 déclarés dans la commande "RESTORE", le terminal accède directement à l'adresse mémoire où sont stockés les paramètres relatifs à l'objet "id[i]" (ou aux objets "id[nbIds]").

25 De la même manière, la commande d'effacement "CLEAN" est déclarée, par exemple par l'entier "8" dans les quatre premiers bits du flux binaire. La zone mémoire dans laquelle il faut effacer les données est identifiée, de la même manière, par les paramètres "serviceID" et "groupID".

A titre d'exemple, on indique qu'une application possible des commandes de stockage au sens de l'invention peut être la suivante. Un utilisateur du terminal qui vit dans une ville telle que Rennes (France) souhaite, par exemple pour  
5 une application météorologique, savoir quelles sont les conditions climatiques prévues dans les jours futurs, dans cette même ville de Rennes. Ainsi, lorsque l'utilisateur du terminal sélectionne la ville "Rennes", sur son terminal, par exemple par le biais d'une commande  
10 interactive, l'information relative à l'intérêt que porte l'utilisateur pour Rennes est transmise au serveur. Le serveur indique, par le biais de la commande "SAVE", que le terminal doit stocker les paramètres d'identification du nœud "Rennes", étant donnée que cette ville est  
15 susceptible de présenter un intérêt pour la suite, pour l'utilisateur du terminal. On indique que le serveur a stocké aussi une information relative au fait qu'il a envoyé la commande "SAVE" à ce terminal. Ainsi, le terminal est identifié dans une mémoire du serveur, en  
20 correspondance de la commande "SAVE". Cette mesure permet au serveur d'envoyer ensuite au terminal une commande "RESTORE", lorsqu'il transmet des informations futures, relatives à la ville de Rennes pour, par exemple, une application météorologique.

25

Ainsi, à l'aide de commandes aussi simples que celles représentées en annexe (dite de "*bas-niveau*"), le serveur n'a pas besoin de transmettre systématiquement les paramètres relatifs au nœud (ou objet de pages multimédia)  
30 que consulte habituellement l'utilisateur, et, plus généralement. Ceux-ci sont déjà stockés par la commande

"SAVE" dans la mémoire du terminal de cet utilisateur. Le serveur envoie simplement une commande "RESTORE" pour récupérer ces paramètres.

5 Habituellement, la plupart des scènes graphiques nécessitent une représentation sous forme d'une liste de primitives (fonctions de bas-niveau) de rendu graphique. A chacune de ces primitives correspond une seule fonction, simple, d'attribution d'un paramètre d'édition graphique.  
10 Les fonctions de stockage, au sens de l'invention, se présente avantageusement comme des fonctions de bas-niveau. Une représentation bas-niveau des fonctions de stockage permet notamment d'avoir une interaction fine avec les objets de l'animation et un transport binaire  
15 entre le serveur et le terminal.

On se réfère maintenant à la figure 3 pour décrire un modèle de transmission et de rendu de scènes graphiques.

20 Une pluralité de modules 42 (décodeurs d'images divers), 43 (gestion de protocoles réseaux), 44 (gestion de polices (ou caractères) de texte), sont stockés en mémoire 41 du terminal TER, en tant que résidents mémoire. En outre, un logiciel client 45, dit "PLAYER", est aussi stocké en tant  
25 que résident mémoire dans la mémoire 41.

Le PLAYER 45 permet de visualiser des contenus animés, interactifs et multimédia sur le terminal mobile. Essentiellement, ce logiciel télécharge ou lit des  
30 informations qui décrivent l'agencement spatio-temporel d'objets graphiques, la façon dont ils sont synchronisés



et les interactions de l'utilisateur du terminal qui sont possibles sur ce contenu.

Le PLAYER 45 interprète alors les interactions de l'utilisateur et en déduit les comportements appropriés des objets graphiques ou les requêtes à effectuer sur le serveur de contenus. Le PLAYER 45 (par exemple de type "STREAMEZZO") inclut des fonctions de rendu d'objets graphiques et des moteurs pour la visualisation (moteur de rendu graphique 50) et l'interaction (moteur d'interaction 51) avec la scène multimédia. Le PLAYER 45 utilise les modules 42, 43, 44 en tant qu'API ("Application Program Interface") du système du terminal mobile qui permettent :

- de décoder les images (module 42),
- de récupérer un flux venant du réseau ou d'une source locale (module 43), et
- de gérer l'affichage du texte et notamment les polices résidentes en standard dans le terminal mobile (module 44).

20

Les contenus pour l'édition comprennent des graphiques vectoriels animés, du son, de la vidéo et des interactions utilisateur. La visualisation de contenus multimédia interactifs, dans des environnements mobiles, nécessite habituellement l'utilisation des techniques de compression afin d'assurer une mise à disposition efficace du contenu et une optimisation du rendu des objets graphiques composant ce contenu. Ce contenu peut alors être visualisé dans de bonnes conditions sur les terminaux mobiles. Les informations lues ou téléchargées par le PLAYER 45 sont donc fortement compressées et le PLAYER doit donc

30

décompresser ces données et les interpréter à la volée pour jouer le contenu.

5 Plus particulièrement, le PLAYER 45 comporte des modules de décodage audio et vidéo, respectivement 46 et 47, un analyseur de flux décodé 49, un gestionnaire de média 48, un moteur de rendu (graphique ou sonore) 50 et un moteur d'interactivité 51.

10 Le fonctionnement du logiciel PLAYER peut se décrire selon les étapes suivantes :

- saisie des données d'entrée via une connexion réseau ou une lecture de fichier,
- décompression de ces données afin d'obtenir une  
15 description des objets graphiques directement utilisables par le moteur de rendu audio et graphique 50,
- composition des objets graphiques entre eux pour créer une scène graphique,
- 20 - rendu graphique proprement dit des objets audio et graphiques, par affichage d'objets visuels ou par jeu d'un son,
- prise en compte des interactions utilisateurs, par exemple un clique d'un pointeur, ou la pression d'une  
25 touche, ou autres,
- établissement d'une connexion à une source d'information locale ou distante si nécessaire.

30 Suite à une requête de l'utilisateur, cette dernière étape va consister en l'ouverture d'une connexion vers le serveur SER et récupérer un flux binaire. Ce flux binaire

est analysé par le PLAYER 45 qui crée alors un objet *ScèneGraph* contenant les différents objets de la scène sous forme de nœuds d'un graphe. Le flux d'information est découpé en paquets qui regroupent des informations qui, dans une réalisation préférée, ne sont valides uniquement qu'à un instant donné et correspondent à un seul type d'information (principe de l'"AccessUnit" du standard MPEG-4).

Le PLAYER analyse chaque paquet et exécute les commandes qui y sont décrites suivant son horloge (non représentée sur les figures), laquelle fournit le temps de la scène multimédia.

Ainsi, on comprendra que ce PLAYER 45, notamment avec les autres modules 42, 43 et 44 du terminal, peut assurer le processus de réception et de décodage de fonctions de stockage d'agencement spatio-temporel d'objets graphiques intervenant dans les pages multimédia incluant par exemple des scènes graphiques et/ou des contenus sonores. Ces fonctions de stockage au sens de la présente invention permettent de gérer la représentation des objets et/ou la modification de leur agencement. Ainsi, les fonctions de stockage permettent de lier plusieurs scènes graphiques et/ou sonores en un service multimédia composite.

On indique que le procédé de l'invention, très général, peut s'appliquer à pratiquement toutes les descriptions d'animations graphiques actuelles, telles que MPEG-4BIFS, SVG, ou autres, dès lors qu'une représentation des signaux

qui composent une application sous forme d'un agencement spatio-temporel d'objets graphiques est prévue.

ANNEXE

```
Save {  
    const bit(4) 9;  
5    uint(12) serviceID;  
    uint(8) groupID;  
    uint(lenBits) nbIds;  
    for (int i = 0; i < nbIds; i++) {  
10        uint(idBits) id[i];  
    }  
}  
  
Restore {  
15    const bit(4) 10;  
    uint(12) serviceID;  
    uint(8) groupID;  
}  
  
20  
Clean {  
    const bit(4) 8;  
    uint(12) serviceID;  
    uint(8) groupID;  
25 }
```

### Revendications

1. Procédé d'édition de pages multimédia auprès d'un terminal, dans lequel un serveur distribue, à un ou  
5 plusieurs terminaux, une partie au moins desdites pages multimédia sous forme d'instructions d'agencements d'objets identifiés par des paramètres associés, caractérisé en ce que le procédé comporte :
- a) au moins une étape préalable pendant laquelle le  
10 serveur transmet au moins une partie des paramètres associés à un objet, ainsi qu'une instruction de stockage desdits paramètres dans une mémoire du terminal,
- b) et au moins une étape courante pendant laquelle le  
15 serveur transmet une instruction de simple restauration desdits paramètres préalablement stockés en mémoire du terminal, pour éditer au moins une page multimédia dans laquelle intervient ledit objet.
2. Procédé selon la revendication 1, caractérisé en ce que  
20 l'étape b) seule est réitérée pour l'édition de plusieurs pages multimédia dans lesquelles intervient ledit objet.
3. Procédé selon l'une des revendications 1 et 2, caractérisé en ce que lesdits paramètres comportent au  
25 moins des attributs déclaratifs d'un agencement de l'objet dans une page multimédia.
4. Procédé selon la revendication 3, caractérisé en ce que  
30 lesdits paramètres comportent en outre un identifiant d'une zone mémoire du terminal allouée pour le stockage desdits attributs.

5. Procédé selon la revendication 4, caractérisé en ce que l'instruction de restauration comporte l'identifiant de ladite zone mémoire pour retrouver lesdits attributs.

5

6. Procédé selon l'une des revendications précédentes, caractérisé en ce qu'il comporte en outre une étape de fin d'édition de pages multimédia incluant ledit objet, étape dans laquelle le serveur transmet au terminal une instruction d'effacement des paramètres associés audit objet.

10

7. Procédé selon la revendication 6, prise en combinaison avec la revendication 5, caractérisé en ce que l'instruction d'effacement comporte l'identifiant de ladite zone mémoire du terminal pour effacer de ladite zone mémoire les paramètres associés à l'objet.

15

8. Procédé selon l'une des revendications précédentes, caractérisé en ce que lesdites instructions sont transmises par paquets du serveur vers le terminal.

20

9. Procédé selon l'une des revendications précédentes, caractérisé en ce que lesdites instructions sont réalisées sous la forme de commandes correspondant à des fonctions de bas-niveau.

25

10. Procédé selon l'une des revendications précédentes, caractérisé en ce que le terminal est un terminal mobile agencé pour coopérer avec un réseau cellulaire.

30

11. Procédé selon l'une des revendications 1 à 10, caractérisé en ce que ledit objet est un objet graphique comportant au moins l'un des éléments parmi :

- une image,
- 5 - une séquence d'images,
- une séquence d'images synthétiques 2D,
- et une séquence d'images synthétiques 3D.

12. Produit programme sous forme de code informatique, caractérisé en ce qu'il comporte une instruction de stockage, dans une mémoire d'un terminal, de paramètres d'au moins un objet destiné à être agencé, en fonction desdits paramètres, dans une page multimédia propre à être éditée auprès dudit terminal.

15

13. Produit programme sous forme de code informatique, caractérisé en ce qu'il comporte une instruction de restauration de paramètres préalablement stockés dans une mémoire d'un terminal, ces paramètres étant associés à au moins un objet destiné à être agencé, en fonction desdits paramètres, dans une page multimédia propre à être éditée auprès dudit terminal.

20

14. Produit programme sous forme de code informatique, caractérisé en ce qu'il comporte une instruction d'effacement de paramètres préalablement stockés dans une mémoire d'un terminal et associés à au moins un objet propre à être agencé, en fonction desdits paramètres, dans une page multimédia éditée auprès dudit terminal.

25

30



15. Signal comportant un code informatique, caractérisé en ce que le code comporte une instruction de stockage, dans une mémoire d'un terminal, de paramètres d'au moins un objet destiné à être agencé, en fonction desdits paramètres, dans une page multimédia propre à être éditée auprès dudit terminal.

16. Signal comportant un code informatique, caractérisé en ce que le code comporte une instruction de restauration de paramètres préalablement stockés dans une mémoire d'un terminal, ces paramètres étant associés à au moins un objet destiné à être agencé, en fonction desdits paramètres, dans une page multimédia propre à être éditée auprès dudit terminal.

17. Signal comportant un code informatique, caractérisé en ce que le code comporte une instruction d'effacement de paramètres préalablement stockés dans une mémoire d'un terminal et associés à au moins un objet propre à être agencé, en fonction desdits paramètres, dans une page multimédia éditée auprès dudit terminal.

1/2

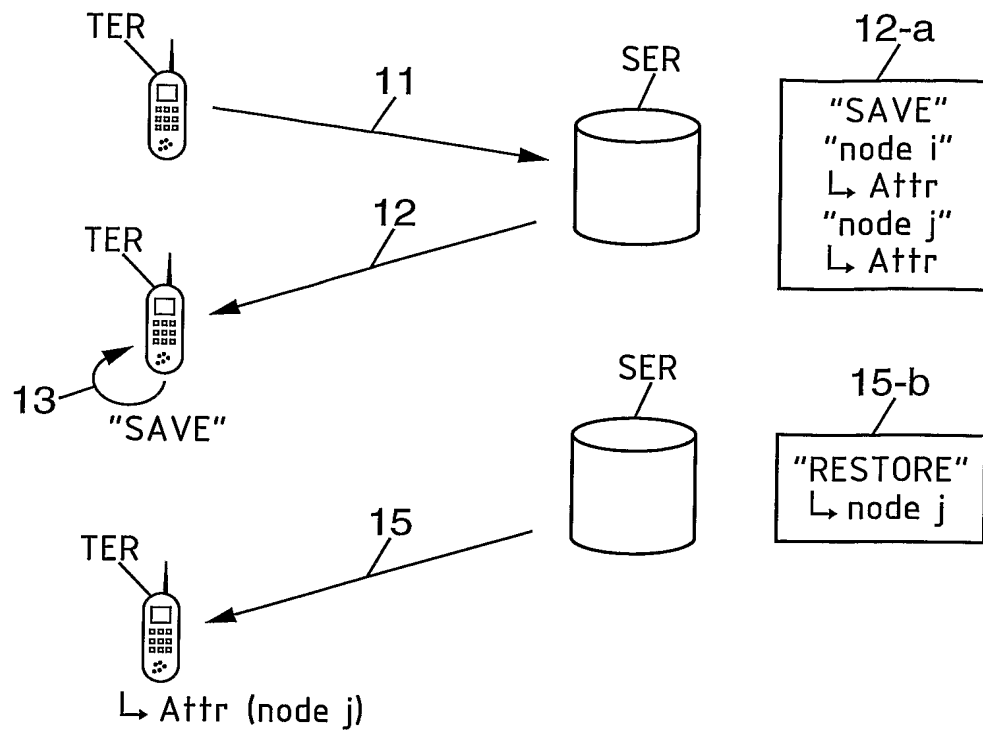


FIG. 1

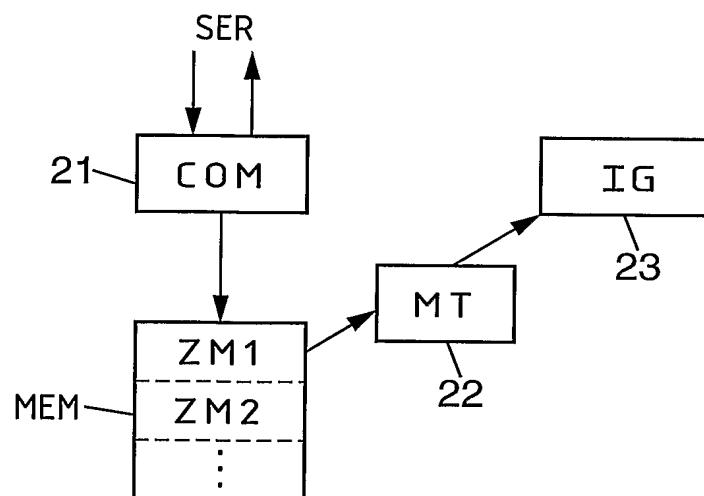


FIG. 2

2/2

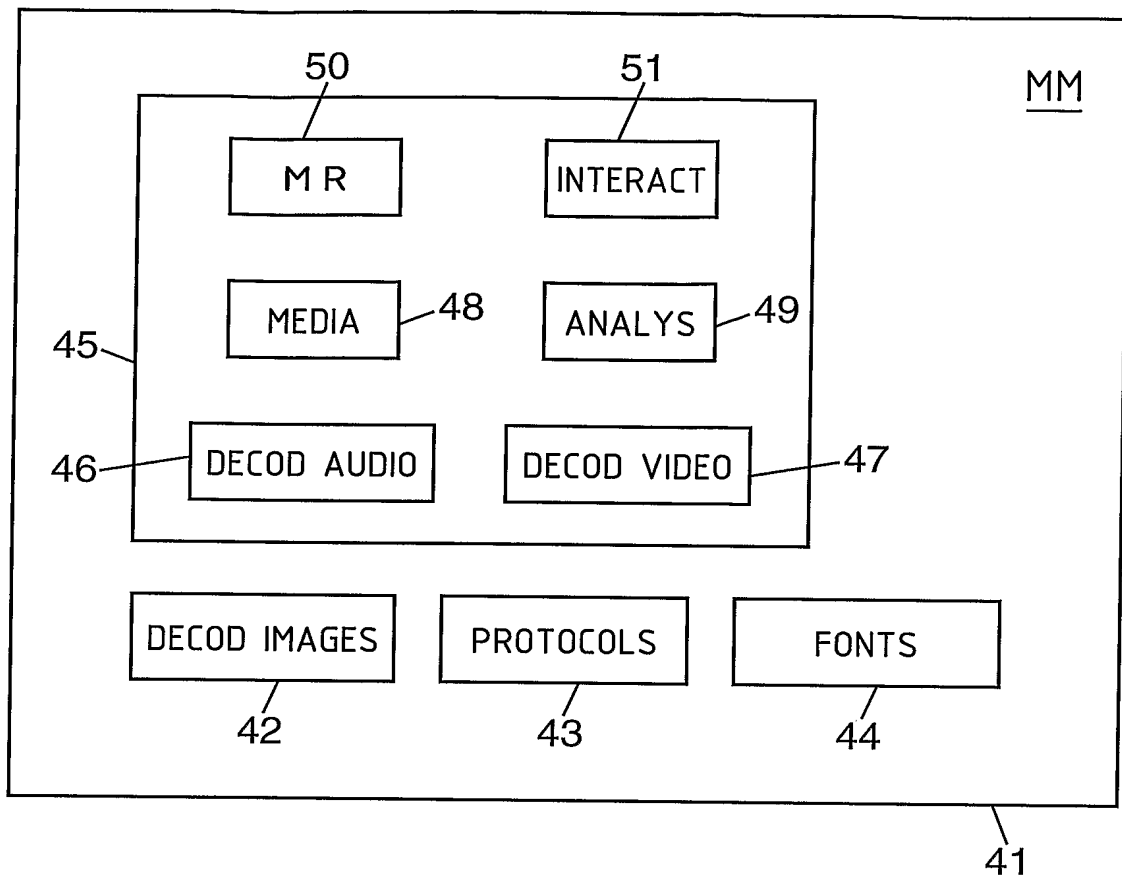


FIG. 3

## INTERNATIONAL SEARCH REPORT

International Application No.

PCT/FR2004/000340

## A. CLASSIFICATION OF SUBJECT MATTER

IPC 7 H04L29/06 H04N7/26 G06F17/30

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC 7 G06F G06T H04N H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal, INSPEC, COMPENDEX, WPI Data

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	DE 100 47 338 A (SIEMENS AG) 18 April 2002 (2002-04-18) the whole document	1-17
X	NIEDERMEIER U ET AL: "An MPEG-7 tool for compression and streaming of XML data" PROCEEDINGS 2002 IEEE INTERNATIONAL CONFERENCE ON MULTIMEDIA AND EXPO, vol. 1, 26 August 2002 (2002-08-26), pages 521-524, XP010604420 abstract section 2.2	1-17



Further documents are listed in the continuation of box C.



Patent family members are listed in annex.

## ° Special categories of cited documents :

- "A" document defining the general state of the art which is not considered to be of particular relevance
- "E" earlier document but published on or after the international filing date
- "I" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- "O" document referring to an oral disclosure, use, exhibition or other means
- "P" document published prior to the international filing date but later than the priority date claimed

- "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
- "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
- "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.
- "&" document member of the same patent family

Date of the actual completion of the international search

18 October 2004

Date of mailing of the international search report

25/10/2004

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2  
NL - 2280 HV Rijswijk  
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,  
Fax: (+31-70) 340-3016

Authorized officer

Bouchaâla, N

## INTERNATIONAL SEARCH REPORT

International Application No

PCT/FR2004/000340

## C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	<p>GIRARDOT M ET AL: "Efficient representation and streaming of XML content over the Internet medium" MULTIMEDIA AND EXPO, 2000. ICME 2000. 2000 IEEE INTERNATIONAL CONFERENCE ON NEW YORK, NY, USA 30 JULY-2 AUG. 2000, PISCATAWAY, NJ, USA, IEEE, US, 30 July 2000 (2000-07-30), pages 67-70, XP010511404 ISBN: 0-7803-6536-4 the whole document</p> <p>-----</p>	1-17

## INTERNATIONAL SEARCH REPORT

### Information on patent family members

International Application No

PCT/FR2004/000340

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
DE 10047338 A	18-04-2002	DE 10047338 A1	18-04-2002

# RAPPORT DE RECHERCHE INTERNATIONALE

Demande Internationale No  
FR/FR2004/000340

<b>A. CLASSEMENT DE L'OBJET DE LA DEMANDE</b> CIB 7    H04L29/06    H04N7/26    G06F17/30		
Selon la classification internationale des brevets (CIB) ou à la fois selon la classification nationale et la CIB		
<b>B. DOMAINES SUR LESQUELS LA RECHERCHE A PORTE</b> Documentation minimale consultée (système de classification suivi des symboles de classement) CIB 7    G06F    G06T    H04N    H04L		
Documentation consultée autre que la documentation minimale dans la mesure où ces documents relèvent des domaines sur lesquels a porté la recherche		
Base de données électronique consultée au cours de la recherche internationale (nom de la base de données, et si réalisable, termes de recherche utilisés) EPO-Internal, INSPEC, COMPENDEX, WPI Data		
<b>C. DOCUMENTS CONSIDERES COMME PERTINENTS</b>		
Catégorie °	Identification des documents cités, avec, le cas échéant, l'indication des passages pertinents	no. des revendications visées
X	DE 100 47 338 A (SIEMENS AG) 18 avril 2002 (2002-04-18) le document en entier -----	1-17
X	NIEDERMEIER U ET AL: "An MPEG-7 tool for compression and streaming of XML data" PROCEEDINGS 2002 IEEE INTERNATIONAL CONFERENCE ON MULTIMEDIA AND EXPO, vol. 1, 26 août 2002 (2002-08-26), pages 521-524, XP010604420 abstract section 2.2 ----- <div style="text-align: right;">-/--</div>	1-17
<div style="display: flex; justify-content: space-between;"> <span><input checked="" type="checkbox"/> Voir la suite du cadre C pour la fin de la liste des documents</span> <span><input checked="" type="checkbox"/> Les documents de familles de brevets sont indiqués en annexe</span> </div>		
° Catégories spéciales de documents cités:		
<div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>*A* document définissant l'état général de la technique, non considéré comme particulièrement pertinent</p> <p>*E* document antérieur, mais publié à la date de dépôt international ou après cette date</p> <p>*L* document pouvant jeter un doute sur une revendication de priorité ou cité pour déterminer la date de publication d'une autre citation ou pour une raison spéciale (telle qu'indiquée)</p> <p>*O* document se référant à une divulgation orale, à un usage, à une exposition ou tous autres moyens</p> <p>*P* document publié avant la date de dépôt international, mais postérieurement à la date de priorité revendiquée</p> </div> <div style="width: 45%;"> <p>*T* document ultérieur publié après la date de dépôt international ou la date de priorité et n'appartenant pas à l'état de la technique pertinent, mais cité pour comprendre le principe ou la théorie constituant la base de l'invention</p> <p>*X* document particulièrement pertinent; l'invention revendiquée ne peut être considérée comme nouvelle ou comme impliquant une activité inventive par rapport au document considéré isolément</p> <p>*Y* document particulièrement pertinent; l'invention revendiquée ne peut être considérée comme impliquant une activité inventive lorsque le document est associé à un ou plusieurs autres documents de même nature, cette combinaison étant évidente pour une personne du métier</p> <p>*Z* document qui fait partie de la même famille de brevets</p> </div> </div>		
Date à laquelle la recherche internationale a été effectivement achevée		Date d'expédition du présent rapport de recherche internationale
18 octobre 2004		25/10/2004
Nom et adresse postale de l'administration chargée de la recherche internationale Office Européen des Brevets, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016		Fonctionnaire autorisé  <div style="text-align: center; font-weight: bold;">Bouchaâla, N</div>

# RAPPORT DE RECHERCHE INTERNATIONALE

Demande Internationale No

PCT/FR2004/000340

## C.(suite) DOCUMENTS CONSIDERES COMME PERTINENTS

Catégorie °	Identification des documents cités, avec, le cas échéant, l'indication des passages pertinents	no. des revendications visées
X	<p>GIRARDOT M ET AL: "Efficient representation and streaming of XML content over the Internet medium"  MULTIMEDIA AND EXPO, 2000. ICME 2000. 2000  IEEE INTERNATIONAL CONFERENCE ON NEW YORK,  NY, USA 30 JULY-2 AUG. 2000, PISCATAWAY,  NJ, USA, IEEE, US,  30 juillet 2000 (2000-07-30), pages 67-70,  XP010511404  ISBN: 0-7803-6536-4  le document en entier  -----</p>	1-17



### Renseignements r

**ix membres de familles de brevets**

Demande Internationale No

ICT/FR2004/000340

Document brevet cité au rapport de recherche	Date de publication	Membre(s) de la famille de brevet(s)	Date de publication
DE 10047338 A	18-04-2002	DE 10047338 A1	18-04-2002